

TITLE OF THE INVENTION

ADVANCED DCT-BASED VIDEO ENCODING METHOD AND APPARATUS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority from Korean Patent Application No. 2002-85447 filed on 27 December 2002, in the Korean Intellectual Property Office, the disclosure of which is incorporated herein in its entirety by reference.

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0002] The present invention relates to a method and apparatus to encode a motion image, and more particularly, to a method and apparatus to remove noise distortion effectively from an image input to a discrete cosine transform (DCT)-based motion image encoder.

2. Description of the Related Art

[0003] A setup box has been developed that receives analog terrestrial broadcast content and encodes it using compression techniques such as MPEG 2 and MPEG 4. However, in the case of terrestrial broadcast content, images received at a receiving site are likely to be distorted due to channel noise. For instance, white Gaussian noise may be contained in an entire image. If such an image is compressed without removal of the noise from the image, compression efficiency is lowered due to the noise.

[0004] Accordingly, significant research has been conducted to remove noise from motion images. Conventionally, noise is removed from an image using a spatial noise reduction method or a temporal noise reduction method.

[0005] A conventional noise removal method will now be explained with reference to FIGS. 1 through 4. FIG. 1 is a block diagram showing a general encoder that encodes motion images. To conduct video-on-demand (VOD) services or motion image communication, the encoder produces a compressed bit stream containing the related data.

[0006] A discrete cosine transform (DCT) unit 110 performs a DCT operation on input image data in units of 8×8 pixel blocks to remove spatial correlation. A quantization (Q) unit 120

performs quantization on a DCT coefficient obtained by the DCT unit 110 to accomplish high efficient loss compression.

[0007] An inverse quantization (IQ) unit 130 inversely quantizes image data quantized by the Q unit 120. An inverse DCT (IDCT) unit 140 performs IDCT on image data that is inversely quantized by the IQ unit 130. After the IDCT unit 140 performs IDCT on the image data, a frame memory 150 stores the image data on a frame-by-frame basis.

[0008] A motion estimation/motion compensation (ME/MC) unit 160 estimates a motion vector (MV) for individual macro blocks and a sum of absolute difference (SAD) corresponding to a block matching error, based on an input current image data frame and a previous image data frame that is stored in the frame memory 150.

[0009] A variable length coding (VLC) unit 170 removes statistical redundancy from DCT-ed and quantized data using the MV estimated by the ME/MC unit 160.

[0010] FIG. 2 is a block diagram illustrating a motion image encoder that uses a conventional noise reduction method. The motion image encoder includes a general video encoder 220 as shown in FIG. 1 and a preprocessor 210. The preprocessor 210 removes noise from an input image using a conventional noise reduction method, and then inputs the input image to the video encoder 220.

[0011] In general, the preprocessor 210 rejects noise from an image using a spatial noise reduction method or a temporal noise reduction method. A spatial-temporal noise reduction method, which is a combination of these methods, may also be used.

[0012] FIGS. 3 and 4 are diagrams illustrating the spatial noise reduction method. Referring to FIG. 3, an edge selector 310 performs high-pass filtering on the eight spatial noise reduction filtering masks shown in FIG. 4 to obtain filtering outputs, and selects a filtering direction to obtain the smallest filtering output with respect to each mask. A coefficient controller 320 performs low-pass filtering in the selected direction by adjusting low-pass filter coefficients in the selected filtering direction. The greater the number of masks the edge selector 310 uses, the higher the precision of directional detection, but the lower the noise reduction effect. On the contrary, the less number of masks used, the higher the noise reduction effect, but the higher the frequency of edge blurring.

[0013] FIG. 5 is a block diagram illustrating the temporal noise reduction method. Referring to FIG. 5, a motion detector 510 estimates the motion and the noise magnitude in a current image, based on an input image and a previous image stored in the frame memory 530. If the motion detector 510 determines a minor amount of motion and a major amount of noise is contained in the current image, a temporal recursive filter 520 performs strong low-pass filtering on the current image in the direction of a temporal axis. In contrast, if the motion detector 510 determines a major amount of motion and a minor amount of noise are contained in the current image, the temporal recursive filter 520 outputs the current image without low-pass filtering. The temporal noise reduction method is effective in processing still images. The temporal noise reduction method is disclosed in US Patent No. 5,969,777.

[0014] However, even when an image is filtered using a conventional spatial noise reduction filter that sharpens the edge of the image, a blurring effect still occurs. Also, since a conventional temporal noise reduction filter is not appropriate for filtering motion images, excess noise still remains in the images.

SUMMARY OF THE INVENTION

[0015] Additional aspects and advantages of the invention will be set forth in part in the description which follows and, in part, will be obvious from the description, or may be learned by practice of the invention.

[0016] The present invention provides a noise reducing method and apparatus that remove noise from an image by filtering the image in a motion image encoder.

[0017] The present invention also provides a motion image encoding apparatus and method that use such a noise removing method and apparatus.

[0018] According to an aspect of the present invention, a discrete cosine transform (DCT)-based motion image encoding method uses a plurality of modified quantization weight matrices, the method comprising: selecting one of the plurality of modified quantization weight matrices based on noise information from input image data; performing DCT on the input image data; and performing quantization on the DCT input image data using the selected modified quantization weight matrix.

[0019] According to another aspect of the present invention, a DCT-based motion image encoding method comprises: creating a modified quantization weight matrix using noise

information from input image data; performing DCT on the input image data; and performing quantization on the DCT input image data using the modified quantization weight matrix.

[0020] According to yet another aspect of the present invention, a DCT-based motion image encoding apparatus uses a plurality of modified quantization weight matrices and comprises: a modified quantization weight matrix storage unit which stores the plurality of modified quantization weight matrices; a modified quantization weight matrix determination unit which selects one of the plurality of modified quantization weight matrices based on noise information from input image data; a DCT unit which performs DCT on the input image data; and a quantization unit which performs quantization on the DCT transformed data using the selected modified quantization weight matrix.

[0021] According to still another aspect of the present invention, a DCT-based motion image encoding apparatus, comprises a modified quantization weight matrix creation unit which creates a modified quantization weight matrix based on noise information from input image data; a DCT unit which performs DCT on the input image data; and a quantization unit which performs quantization on the DCT transformed data using the created modified quantization weight matrix.

BRIEF DESCRIPTION OF THE DRAWINGS

[0022] These and/or other aspects and advantages of the invention will become apparent and more readily appreciated from the following description of the preferred embodiments, taken in conjunction with the accompanying drawings of which:

FIG. 1 is a block diagram illustrating a conventional Moving Picture Expert Group (MPEG) encoder;

FIG. 2 is a block diagram of conventional motion image encoder using a preprocessor;

FIG. 3 illustrates a conventional spatial noise reduction filter;

FIG. 4 illustrates eight masks on which filtering is performed by the filter of FIG. 3;

FIG. 5 is a block diagram illustrating a conventional temporal noise reduction filter;

FIG. 6 is a block diagram illustrating an approximated generalized Wiener filtering which filters non-zero-mean image data;

FIG. 7 is a block diagram illustrating the approximated generalized Wiener filtering which filters non-zero-mean image data in a discrete cosine transform (DCT) domain;

FIGS. 8A through 8C illustrate the structures of filters used during encoding of an intra block;

FIG. 9 illustrates the structure of a general video encoder used during encoding of an inter block;

FIG. 10 is a block diagram of a motion image encoder according to an embodiment of the present invention; and

FIG. 11 is a block diagram of a motion image encoder according to another embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0023] Reference will now be made in detail to the present preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings, wherein like reference numerals refer to the like elements throughout. The embodiments are described below in order to explain the present invention by referring to the figures.

[0024] Hereinafter, an improved noise reduction method utilized by an embodiment of the present invention will be described with reference to FIGS. 6 through 9.

[0025] Preprocess filtering is a major process included in motion image encoding. Noise contained in an image is removed through the preprocess filtering, thus improving encoding efficiency. To remove noise in an image, conventional preprocess filtering is generally performed in the spatial pixel domain of an encoder, whereas preprocess filtering according to an embodiment of the present invention is carried out in the discrete cosine transform (DCT) domain of an encoder.

[0026] In particular, in embodiments of the present invention, approximated, generalized Wiener filtering is performed to remove noise from an image. The approximated Wiener filtering is realized by performing fast unitary transformation, such as DCT, on an image. Alternatively, another filtering technique may be used to filter an image in the DCT domain.

[0027] FIG. 6 is a block diagram illustrating approximated, generalized Wiener filtering which filters non-zero-mean image data in accordance with an embodiment of the present invention.

[0028] In FIG. 6, v denotes an input image block that contains noise, and \hat{w} denotes a row-ordered column vector of a filtered image block. In general, the input image block v indicates a non-zero mean image block. An average estimation unit 610 estimates a mean \hat{m} of an image

block, and a subtracter 620 subtracts the estimated average \hat{m} from the input image block v and outputs data z as the result.

[0029] Next, the data z is filtered by a filtering unit 630 and the filtering unit 630 outputs filtered data \hat{y} as the filtering result. An adder 640 combines the filtered data \hat{y} and the average \hat{m} of the image block and outputs desired filtered data \hat{o} as the combination result.

[0030] A method to perform generalized Wiener filtering on a zero-mean image will now be described. Generalized Wiener filtering performed on a zero-mean image is expressed as follows:

$$\hat{y} = A^{*T} [AL A^{*T}] Az = A^{*T} \tilde{L} Z \dots (1),$$

wherein $\tilde{L} = AL A^{*T}$, $L = [I + \sigma_n^2 R^{-1}]^{-1}$, $R = E[y y^T]$, $Z = Az$, σ_n^2 denotes a noise variance, and A denotes unitary transformation. In this embodiment, an image is filtered using DCT, and therefore, A is a DCT. Also, $A = (C_8 \otimes C_8)$ when C_8 denotes an 8×8 DCT matrix and \otimes denotes a Kronecker multiplication operator.

[0031] In general, because \tilde{L} is approximately diagonalized in the unitary transformation, Equation (1) may also be expressed as follows:

$$\hat{y} = A^{*T} \hat{Y} \dots (2),$$

wherein $\hat{Y} = \tilde{L} Z \approx [\text{Diag } \tilde{L}] Z$.

When Equation (2) is applied to an 8×8 image block, the following equation is obtained:

$$\tilde{y}(k, l) \approx \tilde{p}(k, l) Z(k, l) \dots (3)$$

$\tilde{p}(k, l)$ may be expressed as follows:

$$\tilde{p}(k, l) \approx \frac{1}{1 + \frac{\sigma_n^2}{\sigma^2} \psi(k, l)^{-1}} \dots (4),$$

wherein $\psi(k, l)$ denotes normalized element values that are placed along a diagonal line of $AR A^{*T}$, and σ^2 denotes a variance of the original image y . In general, because σ^2 is an unknown value, it is replaced with a value which is obtained by subtracting a noise variance from a variance z .

[0032] According to Equation (3), a zero-mean image block undergoes approximated, generalized Wiener filtering by multiplying two-dimensional DCT coefficients by $\tilde{p}(k, l)$. After

determination of filtered data $\hat{y}(m, n)$, a final filtered image is obtained by combining the filtered data $\hat{y}(m, n)$ and an average $\hat{m}(m, n)$.

[0033] Generalized Wiener filtering will now be described with respect to a non-zero image model. FIG. 7 shows the structure of an approximated generalized Wiener filter which filters a non-zero image block in a DCT domain in accordance with an embodiment of the present invention. That is, the filter of FIG. 7 filters image models by performing adding and subtracting operations thereon. Assuming that an average image block is obtained by multiplying an input noise-contained DCT block by $S(k, l)$, as expressed in Equation (5), the filter in FIG. 7 may be reconfigured as shown in FIG. 8.

$$\hat{M}(k, l) = S(k, l) \cdot V(k, l) \dots (5)$$

[0034] On the above assumption, an image block filtered in a DCT domain may be expressed using Equations (3) and (5), as follows:

$$\begin{aligned} \hat{W}(k, l) &= \hat{Y}(k, l) + \hat{M}(k, l) = (\hat{p}(k, l) \bullet (1 - S(k, l)) + S(k, l)) \bullet V(k, l) \\ &= F(k, l) \bullet V(k, l) \end{aligned} \dots (6)$$

$F(k, l)$ may be expressed as follows:

$$F(k, l) = \hat{p}(k, l) \bullet (1 - S(k, l)) + S(k, l) = \frac{1 + S(k, l) \bullet \frac{\sigma_n^2}{\sigma^2} \cdot \frac{1}{\psi(k, l)}}{1 + \frac{\sigma_n^2}{\sigma^2} \bullet \frac{1}{\psi(k, l)}} \dots (7)$$

[0035] Equation (6) reveals that the generalized Wiener filtering may be simplified using only $F(k, l)$ and a multiplying operation. Equation (7) discloses that $F(k, l)$ is determined by a signal to noise ratio (SNR), a covariance matrix, and an average matrix.

[0036] In this embodiment, a matrix, which satisfies Equation (5), is selected as an average matrix $S(k, l)$. For example, a DC value of the average matrix $S(k, l)$ in a DCT block may be set forth as follows:

$$S(k,l) = \begin{bmatrix} 10000000 \\ 00000000 \\ 00000000 \\ 00000000 \\ 00000000 \\ 00000000 \\ 00000000 \\ 00000000 \end{bmatrix} \dots (8)$$

[0037] Preprocessing in a motion image encoder will now be described with reference to FIGS. 8 and 9. As mentioned above, even a non-zero input image block may undergo approximated, generalized Wiener filtering by performing a multiplying operation on DCT coefficients.

[0038] FIGS. 8A through 8C are block diagrams illustrating the structures of a motion image encoder in which approximated, generalized Wiener filtering is performed on an intra block. The encoders of FIGS. 8A and 8C filter an intra block in a DCT domain, and perform quantization and variable length coding (VLC) on the filtered intra block without inverse DCT (IDCT).

[0039] In other words, an intra block is simply filtered by multiplying DCT coefficients by $F(k,l)$, and then, each of the DCT coefficients is multiplied (or divided) by a characteristic value in a quantization table. Accordingly, the multiplication and quantization are combined to make a process.

[0040] As is illustrated in FIG. 9, the approximated generalized Wiener filtering shown in FIGS. 8A through 8C may also be performed on an inter block, assuming that noise is removed from motion-compensated estimation block information $p(m,n)$.

[0041] In general, a covariance $\psi(k,l)$ is determined depending on whether an input image block is an inter block or an intra block. Thus, $F(k,l)$ expressed in Equation (7) is also changed according to a block type.

[0042] Hereinafter, a method of calculating an estimated variance for an intra or inter block from which an average is subtracted, will be explained with reference to Equation (9). Where S denotes an $N \times N$ block from which an average is subtracted ($N=8$), a variance matrix of the block may be obtained as follows:

$$\hat{C} = \frac{1}{2} \frac{N}{\text{trace}(S^T S)} (S^T S + S S^T) \dots (9)$$

[0043] Equation (9) is disclosed in an article entitled "Covariance Analysis of Motion-Compensated Frame Differences" listed in IEEE Trans. Circ. Syst. for Video Technol., W. Niehsen and M. Brunig, June 1999.

[0044] An estimated variance may be calculated by applying Equation (9) to a plurality of images. In the case of an intra block, the original image is processed on an 8×8 block-by-block basis. In the case of an inter block, inter blocks are detected and collected using a full search algorithm, and an estimated variance is calculated using Equation (9).

[0045] Next, $R = E[y y^T]$ is calculated using the calculated estimated variance and DCT is performed on R to obtain $\psi = AR A^{*T}$.

[0046] A method of calculating σ_n^2 / σ^2 expressed in Equation (7) will now be described. First, a noise variance σ_n^2 in σ_n^2 / σ^2 may be calculated using a noise estimation unit. Also, when noise and the original image pixels are random variables, the variables are considered independent. Thus, an estimated value $\hat{\sigma}^2$ of a variance σ^2 of the original image is calculated as follows:

$$\hat{\sigma}^2 = \max(\hat{\sigma}_z^2 - \hat{\sigma}_n^2, 0) \dots (10),$$

[0047] wherein σ_z^2 denotes a variance of a macro block (MB). A general motion image encoder calculates the variance σ_z^2 on an MB-by-MB basis. In this embodiment, 8×8 blocks included in the same MB are considered to have the same variances, thus reducing additional computation for variances of the respective 8×8 blocks.

[0048] FIG. 10 illustrates a motion image encoder using a noise reduction method according to an embodiment of the present invention. Hereinafter, the motion image encoder of FIG. 10 using a noise reduction method will be described with reference to FIGS. 1 through 10.

[0049] Comparing the general encoder shown in FIG. 1 to the encoder in FIG. 10, the motion image encoder, in FIG. 10, according to an embodiment of the present invention further includes a noise estimation unit 1080, a quantization (Q) weight matrix determination unit 1092, and a Q weight matrix storage unit 1094. A DCT unit 1010, an IDCT unit 1040, a frame memory 1050, a

motion estimation (ME)/motion compensation (MC) unit 1060, and a variable length coding (VLC) unit 1070 have the same functionality as corresponding units of the encoder shown in FIG. 1, and therefore, their descriptions will be omitted here.

[0050] The Q weight matrix determination unit 1092 determines a quantization (Q) weight matrix, using noise variances σ_n^2 and σ_z^2 which are transmitted from the noise estimation unit 1080 and the ME/MC unit 1060, respectively. Also, the Q weight matrix determination unit 1092 transmits index information regarding the Q weight matrix to the Q weight matrix storage unit 1094.

[0051] Hereinafter, the operation of the Q weight matrix determination unit 1092 will be described in detail. The Q weight matrix determination unit 1092 determines a Q weight matrix with noise variances σ_n^2 and σ_z^2 which are transmitted from the noise estimation unit 1080 and the ME/MC unit 1060, respectively.

[0052] $F(k,l)$ is calculated by Equation (7) in connection with Equation (8) and FIGS. 8 and 9. Next, as shown in FIG. 8C, each of the DCT coefficients $V(k,l)$ of an 8×8 image block is multiplied by $F(k,l)$ to obtain $\hat{W}(k,l)$, and $\hat{W}(k,l)$ is divided by a Q weight matrix in a quantization process.

[0053] A motion image encoder according to an embodiment of the present invention performs an integrated process in which each of the DCT coefficients $V(k,l)$ is multiplied by $F(k,l)$ to obtain $\hat{W}(k,l)$ and $\hat{W}(k,l)$ is divided by a Q weight matrix. If a position component of (k,l) for the Q weight matrix QT is $Q(k,l)$, a position component of (k,l) for the new Q weight matrix QT' is $Q(k,l)/F(k,l)$.

[0054] In this embodiment, the multiplication and division processes are combined to make a process. That is, a plurality of F matrices are predetermined using variances σ_n^2 and σ_z^2 , and new Q weight matrices QT' are calculated using the plurality of F matrices and stored in the Q weight matrix storage unit 1094.

[0055] In this embodiment, five new Q weight matrices QT' are calculated using the variances σ_n^2 and σ_z^2 and stored in the Q weight matrix storage unit 1094. If the variances σ_n^2 and σ_z^2 are determined, σ_n^2 / σ_z^2 may be calculated with these variances, using Equation (10).

[0056] Equation (7) reveals that $F(k,l)$ is determined by $S(k,l)$, $\psi(k,l)$, and σ_n^2 / σ^2 . $S(k,l)$ is calculated using Equation (8). $\psi(k,l)$ changes according to the block type, depending on whether an input image is an inter block or an intra block. Here, σ_n^2 / σ^2 is only a variable used in determining $F(k,l)$. In this embodiment, σ_n^2 / σ^2 are classified for five cases, and five new Q weight matrices QT' are created and stored in the Q weight matrix storage unit 1094.

[0057] The Q weight matrix determination unit 1092 quantizes σ_n^2 / σ^2 using the variances σ_n^2 and σ_z^2 which are input from the noise estimation unit 1080 and the ME/MC unit 1060, respectively. Then, the Q weight matrix determination unit 1092 transmits a new Q matrix index to the Q weight matrix storage unit 1094 corresponding to the quantized σ_n^2 / σ^2 . If the Q weight matrices stored in the Q weight matrix storage unit 1094 are categorized into five cases based on σ_n^2 / σ^2 , the quantization process is performed on five levels and the Q matrix index is a value from 0 to 4.

[0058] When an image contains a large amount of noise, i.e., it has a greater noise variance, σ_n^2 / σ^2 of each of the image blocks with small variances becomes greater. The greater σ_n^2 / σ^2 , the more $F(k,l)$ approximates 0, and the higher the frequency of image blocking, as is well known from Equations (7) and (8). To prevent the blocking of the image, T_{cutoff} is used as shown in the following equation:

$$\sigma_n^2 / \sigma^2 = \min(T_{cutoff}, \sigma_n^2 / \sigma^2) \dots (11)$$

In general, a value of T_{cutoff} is set between 1 and 2.

[0059] The Q weight matrix storage unit 1094 selects a desired Q weight matrix based on Q weight matrix index information input from the Q weight matrix determination unit 1092 and transmits the Q weight matrix to a quantization (Q) unit 1020.

[0060] The Q unit 1020 receives the Q weight matrix and performs quantization using the Q weight matrix.

[0061] An inverse Q unit 1030 performs inverse quantization based on the default Q weight matrix.

[0062] New Q weight matrices may be arbitrarily determined by a user. The embodiment also describes removal of noise from a component Y of an input image block in a DCT domain. However, there is no restriction to which a component of the image block is selected from which noise is removed. For example, a component U or V may be selected when an appropriate additional weight matrix is provided.

[0063] FIG. 11 illustrates a motion image encoder using a noise reduction method, according to another embodiment of the present invention. Comparing FIG. 1 to FIG. 11, the motion image encoder of FIG. 11 further includes a noise estimation unit 1180 and a modified quantization (Q) weight matrix creation unit 1190. In FIG. 11, a DCT unit 1110, an IDCT unit 1140, a frame memory 1150, an ME/MC unit 1160, and a VLC unit 1170 have the same functionality as corresponding units shown in FIG. 1, and therefore, their descriptions will be omitted here.

[0064] The modified Q weight matrix creation unit 1190 creates a modified quantization (Q) weight matrix using variances σ_n^2 and σ_z^2 , which are transmitted from the noise estimation unit 1180 and the ME/MC unit 1160, respectively. The modified Q weight matrix is transmitted to the Q unit 1120.

[0065] Then, the Q unit 1120 performs quantization using the modified Q weight matrix input from the Q weight matrix creation unit 1190.

[0066] A method of creating a modified quantization (Q) weight matrix using variances σ_n^2 and σ_z^2 in accordance with an embodiment of the present invention will now be described in detail.

[0067] In connection with Equation (8) and the descriptions related to FIGS. 8 and 9, $F(k,l)$ is calculated using Equation (7). Next, as shown in FIG. 8(c), each of the DCT coefficients $V(k,l)$ for an 8×8 image block is multiplied by $F(k,l)$ to obtain $\hat{W}(k,l)$, and $\hat{W}(k,l)$ is divided by a Q weight matrix in the quantization process.

[0068] The modified Q weight matrix creation unit 1190, according to an embodiment of the present invention, performs an integrated process in which each of the DCT coefficients $V(k,l)$ is multiplied by $F(k,l)$ to obtain $\hat{W}(k,l)$, and $\hat{W}(k,l)$ is divided by a Q weight matrix in the quantization process. That is, multiplication and quantization are combined to create the

integrated process. If a position component of (k,l) of a Q weight matrix QT is $Q(k,l)$, a position component of (k,l) of a new Q weight matrix QT' is $Q(k,l)/F(k,l)$.

[0069] The modified Q weight matrix creation unit 1190 creates a modified Q weight matrix QT' using variances σ_n^2 and σ_z^2 and provides the matrix QT' to the Q unit 1120. The Q unit 1120 performs quantization on the matrix QT' . The inverse Q unit 1130 performs inverse quantization using the original default Q weight matrix.

[0070] While the present invention has been particularly shown and described with reference to exemplary embodiments thereof, it will be understood by those of ordinary skill in the art that various changes in form and details may be made therein without departing from the spirit and scope of the present invention as defined by the following claims. In particular, the present invention may be applied to any motion image encoding apparatus or method using a compression technique such as MPEG-1, MPEG-2, or MPEG-4.

[0071] The invention may also be embodied as computer readable code on a computer readable recording medium. The computer readable recording medium is any data storage device that stores data which is read by a computer system. Examples of the computer readable recording medium include read-only memory (ROM), random-access memory (RAM), CD-ROMs, magnetic tapes, floppy disks, optical data storage devices, and the like. Also, the computer readable codes may be transmitted via a carrier wave such as the Internet. The computer readable recording medium may also be distributed over network-coupled computer systems so that the computer readable code is stored and executed in a distributed fashion.

[0072] As described above, in a motion image encoding apparatus and method according to an embodiment of the present invention, high-performance filtering may be realized by using a noise reduction apparatus in a DCT domain and by adding memory and logic operations to a conventional motion image encoding apparatus and method, respectively. A motion image encoding apparatus and method according to an embodiment of the present invention are compatible with a conventional motion image encoding apparatus and method.

[0073] Although a few embodiments of the present invention have been shown and described, it would be appreciated by those skilled in the art that changes may be made in this embodiment without departing from the principles and spirit of the invention, the scope of which is defined in the claims and their equivalents.